

The price of bandit information in multiclass online classification

Amit Daniely * Tom Helbertal †

February 6, 2013

Abstract

We consider two scenarios of multiclass online learning of a hypothesis class $H \subseteq Y^X$. In the *full information* scenario, the learner is exposed to instances together with their labels. In the *bandit* scenario, the true label is not exposed, but rather an indication whether the learner’s prediction is correct or not. We show that the ratio between the error rates in the two scenarios is at most $8 \cdot |Y| \cdot \log(|Y|)$ in the realizable case, and $\tilde{O}(\sqrt{|Y|})$ in the agnostic case. The results are tight up to a logarithmic factor and essentially answer an open question from [Daniely et al. \(2011\)](#).

We apply these results to the class of γ -margin multiclass linear classifiers in \mathbb{R}^d . We show that the bandit error rate of this class is $\tilde{\Theta}\left(\frac{|Y|}{\gamma^2}\right)$ in the realizable case and $\tilde{\Theta}\left(\frac{1}{\gamma}\sqrt{|Y|T}\right)$ in the agnostic case. This resolves an open question from [Kakade et al. \(2008\)](#).

1 Introduction

Online multiclass classification is an important task in Machine Learning. In its basic form, which we refer as the *full information* scenario, the learner is required to predict the label of a new example, based on previously observed labeled examples. Recently, the *bandit* scenario has received much attention (e.g. [Auer et al. \(2003\)](#), [Kakade et al. \(2008\)](#), [Dani et al. \(2008\)](#), [Auer et al. \(2002\)](#)). Here, the learner does not observe labeled examples, but rather, it observes unlabeled examples, predicts their labels and only receives an indication whether his prediction was correct. The relevance of the bandit scenario to practice is evident – a canonical example is internet advertising, where the advertiser chooses a commercial (which is thought as a label) upon the information it has on the user (which is thought as an instance). After choosing a commercial, the advertiser only knows if the user has clicked the commercial or not.

Let X be the instance space and Y the label space. Denote $k = |Y|$. To evaluate learning algorithms, it is common to compare them to the best hypothesis coming from some fixed

*Department of Mathematics, Hebrew University, Jerusalem 91904, Israel. amit.daniely@mail.huji.ac.il

†Department of Mathematics, Hebrew University, Jerusalem 91904, Israel. tom.halbertal@mail.huji.ac.il

hypothesis class $H \subseteq Y^X$. We define the *error rate* of H as the least number, $\text{Err}_H(T)$, for which *some* algorithm is guaranteed to make at most $\text{Err}_H(T)$ mistakes more than the best hypothesis in H , when running on a sequence of length T . We emphasize that we consider all algorithms, not only efficient ones.

It is clear that learning is harder in the bandits scenario. The purpose of this work is to quantify how larger is the error rate in this scenario. Our main results show that, for every hypothesis class H , the error rate in the bandit scenario is only $\tilde{O}(k)$ times larger in the realizable case (i.e. in the case that some hypothesis H makes no mistakes) and $\tilde{O}(\sqrt{k})$ times larger in the general (agnostic) case. We note that our results hold also for the multiclass multi-label categorization, where a set of labels are allowed to be correct. As an application, we use our results to quantify the error rate of the class of large margin halfspaces classifiers.

1.1 Related Work

Cardinality based vs. Dimension based bounds. The celebrated result of Cesa-Bianchi et al. (1997) shows that, in the full-info scenario, the error rate is upper bounded by $O(\sqrt{\log(|H|)T})$. In the full-info-realizable case, the majority algorithm achieves an error rate of $O(\log(|H|))$.

These two bounds are tight for several hypothesis classes. However, there are several important classes for which much better error rates can be achieved. For example, those bounds are meaningless for infinite hypothesis classes. However, several such classes (e.g. the class of large margin halfspaces classifiers) do admit finite error rate.

The reason to those deficiencies is that the quantity $\log(|H|)$ does not quantify the true complexity of the class, but only upper bounds it. To remedy that, Daniely et al. (2011), following a binary version from Ben-David et al. (2009) and Littlestone (1988), proposed a notion of dimension (a-la VC dimension), called the *Littlestone dimension*. As shown in Daniely et al. (2011), the error rate of H , in the full-info scenario, is $\tilde{\Theta}(\sqrt{L(H)T})$ in the agnostic case and $\Theta(L(H))$ in the realizable case¹.

The results of Daniely et al. (2011) show that the term $\log(|H|)$ in the result of Cesa-Bianchi et al. (1997) and in the bound of the majority algorithm can be replaced by $L(H)$ (the algorithms they use are different, however), leading to a tight (up to log factors of T and k) characterization of the error rate. Our results can be seen as analogues of these results in the bandit scenario: By the algorithm of Auer et al. (2003) the error rate of H in the bandit scenario is $O(\sqrt{kT\log(|H|)})$. By the Majority algorithm, the error rate in the realizable-bandit scenario is $O(k\log(|H|))$. Our results upper bound the error rates by $\tilde{O}(\sqrt{kTL(H)})$ and $\tilde{O}(kL(H))$ respectively.

Since the Littlestone dimension characterizes the full-info error rate, our results imply an upper bound on the ratio between the bandit and full-info error rates. To the best of our knowledge, these are the first upper bounds on this ratio that hold for every class. We note also that since $L(H) \leq \log(|H|)$, our bounds, up to log factors, imply the bounds of Auer et al. (2003) and the majority algorithm.

Comparison to other settings. In the statistical/PAC settings, one assumes that the

¹A detailed study of online analogs to statistical complexity measures can be found in Rakhlin et al. (2010)).

sequence of examples is drawn i.i.d. from some distribution on $X \times Y$. In these settings, it is not hard to show that the bandit error rate is at most $O(k)$ times larger than the full-info rate (see Section 4 and Daniely et al. (2011)). Our results generalize these facts to the adversarial setting.

2 Our Results

2.1 Problem setting and Background

Settings. Fix an instance space X and a label space Y . Denote $Z = X \times Y$, $\mathcal{Y} = 2^Y$, $\mathcal{Z} = X \times \mathcal{Y}$ and $k = |Y|$. We consider two scenarios of multiclass online learning. In the *full information* scenario, at each step $t = 1, 2, \dots$ a *full-info learning algorithm* is exposed to an instance $x_t \in X$, predicts a label $\hat{y}_t \in Y$ and then observes a list of true labels $Y_t \subset Y$ (note that this is little more general than the vanilla multiclass setting in which $|Y_t| = 1$). The prediction \hat{y}_t can be based only on the previously observed labeled examples $(x_1, Y_1), \dots, (x_{t-1}, Y_{t-1})$ and on x_t . The *bandit* scenario is similar. The sole difference is that, after a *bandit learning algorithm* predicts a label, the true labels are not exposed, but only an indication whether the algorithm's prediction was correct or not. Therefore, the prediction \hat{y}_t can be based only on the previously observed unlabeled examples x_1, \dots, x_{t-1}, x_t and on previously obtained indications $1(\hat{y}_1 \in Y_1), \dots, 1(\hat{y}_{t-1} \in Y_{t-1})$. We assume that the choice of the sequence (x_t, Y_t) is adversarial, but the adversary chooses Y_t before the algorithm predicts \hat{y}_t . In particular, the algorithm may choose \hat{y}_t at random after the adversary chose Y_t .

Let H be a hypothesis class, which might be either class of functions from X to Y , or, a class of functions from X to \mathbb{R}^Y . We say that a sequence $(x_1, y_1), \dots, (x_T, y_T) \in Z$ is *realizable* by H if there exists a function $h \in H$ such that either $\forall 1 \leq t \leq T$, $h(x_t) = y_t$, for the case that $H \subset Y^X$, or $\forall 1 \leq t \leq T$, $h_{y_t}(x_t) \geq 1 + \max_{y \neq y_t} h_y(x_t)$, in the case² that $H \subset (\mathbb{R}^Y)^X$. We denote by $H(T) \subset Z^T$ the sequences of length T that are realizable by H . We say that a sequence $(x_1, Y_1), \dots, (x_T, Y_T) \in \mathcal{Z}$ is *realizable by H* if there exist $y_1 \in Y_1, \dots, y_T \in Y_T$ such that the sequence $(x_1, y_1), \dots, (x_T, y_T)$ is realizable by H .

The *error of H on a sequence $\mathbf{z} = ((x_1, Y_1), \dots, (x_T, Y_T)) \in \mathcal{Z}^T$* is minimal number of errors that a hypothesis from H makes on the sequence \mathbf{z} . Namely,

$$\text{Err}(H, \mathbf{z}) = \min_{((x_1, y_1), \dots, (x_T, y_T)) \in H(T)} \sum_{t=1}^T 1(y_t \notin Y_t).$$

Let A be a (either full-info or bandit) learning algorithm. Given $\mathbf{z} \in \mathcal{Z}^T$, we denote by $\text{Err}(A, \mathbf{z})$ the expected number of errors A makes, running on the sequence \mathbf{z} . We define the *realizable error rate* of A w.r.t. H as the worst case performance of A on a length T

²In some contexts it favourable to use a margin-dependent notion of realization. Namely, to define a γ -realizable sequence by requiring that $\forall 1 \leq t \leq T$, $h_{y_t}(x_t) \geq \gamma + \max_{y \neq y_t} h_y(x_t)$. Observing that a sequence is γ -realizable by H iff it is realizable by $\left(\frac{1}{\gamma} \cdot H\right)$, it is easy to interpolate between the two definitions. Our choice of the above definition is merely for the sake of clarity.

realizable sequence, namely,

$$\text{Err}_A^r(T) = \sup_{\mathbf{z} \in \mathcal{Z}^T, \text{Err}(H, \mathbf{z})=0} \text{Err}(A, \mathbf{z}) .$$

The *agnostic error rate* of A is its worst case performance over all length T sequences, namely,

$$\text{Err}_A^a(T) = \sup_{\mathbf{z} \in \mathcal{Z}^T} \text{Err}(A, \mathbf{z}) - \text{Err}(H, \mathbf{z}) .$$

The *realizable and agnostic full-info error rates* of the class H are the best achievable error rates, namely,

$$\text{Err}_H^r(T) = \inf_{A \text{ is a full-info alg.}} \text{Err}_A^r(T) \text{ and } \text{Err}_H^a(T) = \inf_{A \text{ is a full-info alg.}} \text{Err}_A^a(T) .$$

Similarly, the *realizable and agnostic bandit error rates* of the class H are

$$\text{B-Err}_H^r(T) = \inf_{A \text{ is a bandit alg.}} \text{Err}_A^r(T) \text{ and } \text{B-Err}_H^a(T) = \inf_{A \text{ is a bandit alg.}} \text{Err}_A^a(T) .$$

Our main focus is to understand how larger is the error rate in the bandit scenario, compared to the full-info scenario. Thus we define the agnostic and realizable *price of bandit information* of H by

$$\text{POB}_H^r(T) = \frac{\text{B-Err}_H^r(T)}{\text{Err}_H^r(T)} \text{ and } \text{POB}_H^a(T) = \frac{\text{B-Err}_H^a(T)}{\text{Err}_H^a(T)} .$$

The Littlestone dimensions. Daniely et al. (2011) (following Ben-David et al. (2009) and Littlestone (1988)) defined two combinatorial notions of dimension that characterize the error rates of a class H . Let T be a rooted tree whose internal nodes are labeled by X and whose edges are labeled by Y . We say that T is *L-shattered* by H if, for every root-to-leaf path x_1, \dots, x_T , the sequence $(x_1, y_1), \dots, (x_{T-1}, y_{T-1})$, where y_t is the label associated with the edge $x_t \rightarrow x_{t+1}$, is realizable by H . The *Littlestone dimension* of H , denoted $\text{L}(H)$, is the maximal depth of a complete binary tree³ that is L-shattered by H . We say that T is *BL-shattered* by H if, for every root-to-leaf path x_1, \dots, x_T , if y_t is the label associated with $x_t \rightarrow x_{t+1}$, then there exists a realizable sequence $(x_1, y'_1), \dots, (x_{T-1}, y'_{T-1})$ such that $\forall i, y'_i \neq y_t$. The *Bandit Littlestone dimension* of H , denoted $\text{BL}(H)$, is the maximal depth of a complete k -ary tree that is BL-shattered by H .

Theorem 2.1 (Daniely et al. (2011))

- For every class H and for every $T \geq \text{L}(H)$, $\frac{1}{2}\text{L}(H) \leq \text{Err}_H^r(T) \leq \text{L}(H)$ and $\Omega(\sqrt{\text{L}(H)T}) \leq \text{Err}_H^a(T) \leq O(\sqrt{\text{L}(H)T \log(kT)})$.
- For every class H , $\text{B-Err}_H^r(T) \leq \text{BL}(H)$. Moreover, for every deterministic bandit algorithm, $\text{Err}_A^r(T) \geq \min\{T, \text{BL}(H)\}$.

³By a complete binary tree, we mean a tree whose all internal nodes have two children and all leaves are at the same depth.

The class of large-margin multiclass linear separators. Denote by B^d the unit ball in \mathbb{R}^d . We identify every matrix $W \in M_{k \times d}(\mathbb{R})$ with the linear function it defines on B^d (i.e. $x \mapsto Wx$). Denote by $\|W\|_F$ the Frobenius norm of W , namely, $\|W\|_F = \sqrt{\sum_{i=1}^k \sum_{j=1}^d W_{ij}^2}$. For $D > 0$ let $\mathcal{W}^{d,k}(D) = \{W \in M_{k \times d}(\mathbb{R}) : \|W\|_F \leq D\}$.

A multiclass variant of the Perceptron algorithm (see Crammer and Singer (2003)) makes at most $2 \cdot D^2$ mistakes whenever it runs on a sequence that is realizable by $\mathcal{W}^{d,k}(D)$. Therefore, $L(\mathcal{W}^{d,k}(D)) \leq 2 \cdot D^2$. Also, it is not hard to see that $\min\{d, \lfloor D^2 \rfloor\} \leq L(\mathcal{W}^{d,k}(D))$. Thus, we have

$$\min\{d, \lfloor D^2 \rfloor\} \leq L(\mathcal{W}^{d,k}(D)) \leq 2 \cdot D^2 \quad (1)$$

2.2 Results

Our first result bounds the bandit-realizable error rate in terms of the Littlestone dimension.

Theorem 2.2 *For every hypothesis class H , $B\text{-Err}_H^r(T) \leq 4|Y| \log(|Y|) L(H)$.*

Together with Theorem 2.1 we conclude that the bandit-realizable error rate is at most $\tilde{O}(k)$ larger than the full-info-realizable error rate. Namely, for every hypothesis class H ,

$$POB_H^r(T) \leq 8|Y| \cdot \log(|Y|) . \quad (2)$$

It is not hard to see (e.g. by Lemma 2) that for $H = Y^X$ and $T \geq (|Y| - 1) \cdot |X|$, we have that $B\text{-Err}_H^r(T) \geq \frac{(|Y|-1) \cdot |X|}{2} = \frac{(|Y|-1) \cdot L(H)}{2}$. Thus, Theorem 2.2 is tight up to a factor of $\log(|Y|)$. By Theorem 2.1,

$$POB_H^r(T) = \frac{B\text{-Err}_H^r(T)}{Err_H^r(T)} \geq \frac{\frac{(|Y|-1) \cdot L(H)}{2}}{L(H)} = \frac{(|Y|-1)}{2} .$$

Thus, Equation (2) is tight up to a factor of $\log(|Y|)$ as well. In Daniely et al. (2011) it has been asked how large can be the ratio $\frac{BL(H)}{L(H)}$. It can be easily seen that $\frac{BL(H)}{L(H)}$ can be as large as $|Y| - 1$ (this is true, for example, when X is finite and $H = Y^X$). Theorem 2.2, together with Theorem 2.1, shows that $\frac{BL(H)}{L(H)} \leq 4|Y| \log(|Y|)$, essentially answering the question of Daniely et al. (2011).

For the agnostic case we show the following result:

Theorem 2.3 *For every class H , $B\text{-Err}_H^a(T) \leq e \cdot \sqrt{T|Y| L(H) \log(T \cdot |Y|)}$.*

Together with Theorem 2.1, it follows that for every class H ,

$$POB_H^a(T) = O\left(\sqrt{|Y| \cdot \log(|Y| \cdot T)}\right) . \quad (3)$$

Relying on the construction from section 5 of Auer et al. (2003), it is not hard to show that for $H = Y^X$ and $T \geq |Y| \cdot |X| = |Y| \cdot L(H)$ it holds that $B\text{-Err}_H^a(T) \geq \frac{1}{20} \sqrt{L(H) \cdot T \cdot |Y|}$. Thus, together with Theorem 2.1, Theorem 2.3 and Equation (3) are tight up to a logarithmic factor of $\log(|Y| \cdot T)$.

Next, we apply Theorems 2.2 and 2.3 to analyse the bandit error rate of large margin multiclass linear separators.

Theorem 2.4 For every $D > 0$ and $d, k \in \mathbb{N}$

$$\text{Err}_{\mathcal{W}^{d,k}(D)}^r(T) \leq 8 \cdot k \cdot \log(k) \cdot D^2, \quad \text{Err}_{\mathcal{W}^{d,k}(D)}^a(T) \leq 4 \cdot D \cdot \sqrt{T k \log(T \cdot k)}$$

Moreover, for $L = \min\{d, \lfloor D^2 \rfloor\}$ and $T \geq k \cdot L$,

$$\text{Err}_{\mathcal{W}^{d,k}(D)}^r(T) \geq \frac{(k-1) \cdot L}{2}, \quad \text{Err}_{\mathcal{W}^{d,k}(D)}^a(T) \geq \frac{1}{20} \sqrt{LTk}$$

Kakade et al. (2008) have shown an (inefficient) randomized algorithm that makes, w.p. $1 - \delta$, at most $O(k^2 D^2 \ln(\frac{T+k}{\delta}) \cdot (\ln D + \ln \ln(\frac{T+k}{\delta})))$ mistakes, whenever it runs on a sequence $(x_1, Y_1), \dots, (x_T, Y_T)$ that is realizable by $\mathcal{W}^{d,k}(D)$. It has been asked there what is the optimal error rate, and whether there exists an asymptotically finite bound on the error rate that does not depend on the dimension d . Theorem 2.4 answers the second question in affirmative and essentially answers the first question. Also, Kakade et al. (2008) have conjectured that for fixed D and k , the bandit agnostic error rate of $\mathcal{W}^{d,k}(D)$ should be $O(\sqrt{T})$. Theorem 2.4 validates this conjecture, up to a factor of $\sqrt{\log(T)}$.

The bound in Theorem 2.4 is rather tight when the dimension, d , is larger than the complexity D^2 . To complete the picture, we note that in Kakade et al. (2008) it has been shown that $\text{B-Err}_{\mathcal{W}^{d,k}(D)}^r(T) \leq O(k^2 d \log(D))$. Here we show a corresponding lower bound.

Theorem 2.5 For every $D^2 \geq k^3 d$ and $T \geq \frac{dk^2}{8}$, $\text{B-Err}_{\mathcal{W}^{d,k}(D)}^r(T) \geq \lfloor d/2 \rfloor \cdot (k-1) \cdot \frac{k}{4}$.

2.3 Proof techniques

The proof of Theorem 2.2 constitutes most of the technical novelty of the paper. The algorithm we use belongs to the family of “majority vote” algorithms such as the Standard Optimal Algorithms of Littlestone (1988), Ben-David et al. (2009) and Daniely et al. (2011). These algorithms start with a hypothesis class $H_1 = H$. At each step t , they predict the label predicted by “most” hypotheses in H_t , where “most” is quantified in a certain way. After an indication is given for that prediction (i.e., after the true label is exposed in the full-info scenario or after an indication whether the algorithm’s guess was correct or not in the bandit scenario), the algorithm constructs H_{t+1} by throwing away all functions that are in contradiction with that indication.

A notable distinction is that instead of a single hypothesis class, our algorithm keeps a *collection* of hypothesis classes. At each step, each class in that collection is either split, thrown away, or remains untouched. The prediction at each step aims to minimize a measure of capacity for collections of hypothesis classes, which we define. We show that this measure shrinks to 1 after $4|Y| \cdot \log(|Y|) \cdot L(H)$ mistakes. From that point, the algorithm makes no mistakes.

Theorem 2.3 is based on an argument from Ben-David et al. (2009) (see also Daniely et al. (2011)). We represent each class by a relatively small number of experts and apply the result of Auer et al. (2003) on these set of experts. Theorem 2.4 is deduced from Theorems 2.2, 2.3 and Equation (1).

To prove Theorem 2.5, we first consider the class H of all functions $f : [\lfloor \frac{d}{2} \rfloor] \times [k] \rightarrow [k]$ such that $f|_{\{j\} \times [k]}$ is a bijection for every $j \in [\lfloor \frac{d}{2} \rfloor]$. We show that $\text{B-Err}_H^r(T) \geq \lfloor \frac{d}{2} \rfloor \cdot (k-1) \cdot \frac{k}{4}$. Then, we adapt a construction from Daniely et al. (2011) to show that H can be realized by $\mathcal{W}^{d,k}(D)$.

3 Proofs

Throughout, we denote by k the number of labels. We prove Theorems 2.2 and 2.3 for the case that $H \subset Y^X$. The case of real-valued H can be handled along the same lines. We say that a hypothesis class H_1 is *realized* by H_2 , if, $\forall T, H_1(T) \subset H_2(T)$. It is clear that in this case the error rates and the Littlestone dimensions of H_1 are no larger than those of H_2 .

3.1 Theorem 2.2

Let \mathcal{H} be a collection of non-empty subsets of H . We define its *capacity* by $C(\mathcal{H}) = \sum_{V \in \mathcal{H}} k^{2L(V)}$. We note that for $\mathcal{H} = \{H\}$, it holds that $C(\mathcal{H}) = k^{2L(H)}$. Also, for non-empty \mathcal{H} , $C(\mathcal{H}) \geq 1$. Our algorithm starts with $\mathcal{H}_1 = \{H\}$. At each step it modifies \mathcal{H}_t such that (1) $C(\mathcal{H}_t)$ shrinks with every false prediction and (2) all hypotheses that are consistent with the previously observed instances are in one of the subclasses of \mathcal{H}_t .

Given $x \in X$, $y \in Y$ and $V \subset H$ devote $V_x^y = \{f \in V : f(x) = y\}$. For a collection, \mathcal{H} , of subsets of H we define

$$\Lambda(\mathcal{H}, x, y_0) = \{V \in \mathcal{H} : \forall y \neq y_0, L(V_x^y) < L(V)\}$$

$$\lambda(\mathcal{H}, x, y_0) = \{V_x^y : V \in \Lambda(\mathcal{H}, x, y_0), y \neq y_0, V_x^y \neq \emptyset\} \cup \mathcal{H} \setminus \Lambda(\mathcal{H}, x, y_0)$$

$$P_{H,x}(y) = C(\mathcal{H}) - C(\lambda(\mathcal{H}, x, y))$$

Algorithm 1

-
- 1: Set $\mathcal{H}_1 = \{H\}$.
 - 2: **for** $t = 1, 2, \dots$ **do**
 - 3: receive x_t
 - 4: Predict $y \in \text{argmax}_{y \in Y} P_{\mathcal{H}_t, x_t}(y)$.
 - 5: If the prediction is wrong, update $\mathcal{H}_{t+1} = \lambda(\mathcal{H}_t, x_t, y)$. Otherwise, $\mathcal{H}_{t+1} = \mathcal{H}_t$.
 - 6: **end for**
-

Claim 1 Algorithm 1 makes less than $4L(H)k \log(k)$ mistakes.

Proof Fix $V \in \mathcal{H}_t$ and let $y \in \text{argmax}_{y' \in Y} L(V_x^{y'})$. Since there is at most one $y' \in Y$ such that $L(V_x^{y'}) = L(V)$ (see Littlestone (1988)), it follows that for every $y' \neq y$, $L(V_x^{y'}) < L(V)$. In particular, $V \in \Lambda(\mathcal{H}_t, x, y)$ and

$$\begin{aligned} k^{2L(V)} - \sum_{y' | V_x^{y'} \neq \emptyset, y' \neq y} k^{2L(V_x^{y'})} &\geq k^{2L(V)} - (k-1) \cdot k^{2L(V)-2} \\ &= \left(1 - \frac{k-1}{k^2}\right) k^{2L(V)} \\ &\geq \left(1 - \frac{1}{k}\right) k^{2L(V)}. \end{aligned}$$

Thus,

$$\begin{aligned}
\sum_{y \in Y} P_{\mathcal{H}_t, x_t}(y) &= \sum_{y \in Y} \sum_{V \in \Lambda(\mathcal{H}_t, x_t, y)} (k^{2L(V)} - \sum_{y' | V_x^{y'} \neq \emptyset, y' \neq y} k^{2L(V_{x_t}^{y'})}) \\
&\geq \sum_{V \in \mathcal{H}_t} \sum_{y \in Y: V \in \Lambda(\mathcal{H}_t, x_t, y)} (k^{2L(V)} - \sum_{y' | V_x^{y'} \neq \emptyset, y' \neq y} k^{2L(V_{x_t}^{y'})}) \\
&\geq \sum_{V \in \mathcal{H}_t} \left(1 - \frac{1}{k}\right) k^{2L(V)} \\
&= \left(1 - \frac{1}{k}\right) C(\mathcal{H}_t)
\end{aligned}$$

It follows that for some $y \in Y$, $P_{\mathcal{H}_t, x_t}(y) \geq \frac{1}{k} \left(1 - \frac{1}{k}\right) C(\mathcal{H}_t) \geq \frac{1}{2k} C(\mathcal{H}_t)$. Thus, if the algorithm errs at time t then

$$C(\mathcal{H}_t) - C(\mathcal{H}_{t+1}) \geq \frac{1}{2k} C(\mathcal{H}_t) \Rightarrow C(\mathcal{H}_{t+1}) \leq \left(1 - \frac{1}{2k}\right) C(\mathcal{H}_t)$$

It follows that after $4L(H)k \log(k)$ mistakes it will hold that

$$\begin{aligned}
C(\mathcal{H}_t) &\leq \left(1 - \frac{1}{2k}\right)^{4L(H)k \log(k)} C(\mathcal{H}_1) \\
&< e^{-2L(H) \log(k)} k^{2L(H)} = 1
\end{aligned}$$

However, it is not hard to see that each hypothesis which is consistent with the history up to time $t-1$ is in one of the classes of \mathcal{H}_t . As we assume that the sequence is realizable, there is at least one consistent hypothesis. Thus, for every t , $C(\mathcal{H}_t) \geq 1$. It follows that the algorithm makes less than $4L(H)k \log(k)$ mistakes.

□

3.2 Theorem 2.3

We use a result from [Auer et al. \(2003\)](#), which we briefly describe next. Suppose that at each step, t , before the algorithm chooses its prediction, it observes N advices $(f_1^t, \dots, f_N^t) \in Y^N$, which can be used to determine its prediction. We refer to f_i^t as the prediction made by the expert i at time t and denote by $L_{i,T} = |\{t \in [T] : f_{i,t} \notin Y_t\}|$ the loss of the expert i at time T . For every sequence $z \in \mathcal{Z}^T$, the algorithm from [Auer et al. \(2003\)](#), section 7, makes at most $\min_{i \in [N]} L_{i,T} + e\sqrt{kT \log(N)}$ mistakes in expectation.

Suppose that, for every $f \in H$, we construct an expert, E_f , whose advice at time t is $f(x_t)$. Denote by $L_{f,t}$ the loss of the expert E_f at time t . Running the algorithm of [Auer et al. \(2003\)](#) with this set of experts yields an algorithm whose agnostic error rate is at most $e\sqrt{kT \log(|H|)}$. We proceed by imitating this set of experts with a more compact set of experts, which will allow us to bound the loss in terms of $L(H)$ instead of $\log(|H|)$.

Let $A_T = \{A \subset [T] \mid |A| \leq L(H)\}$. For every $A \in A_T$ and $\phi : A \rightarrow Y$, we define an expert $E_{A,\phi}$. The expert $E_{A,\phi}$ imitates the SOA algorithm (Algorithm 2 in the appendix)

when it errs exactly on the examples $\{x_t \mid t \in A\}$ and the true labels of these examples are determined by ϕ . The expert $E_{A,\phi}$ proceeds as follows:

Set $V_1 = H$.

For $t = 1, 2, \dots, T$

Receive x_t .

Set $l_t = \operatorname{argmax}_{y \in Y} L(\{f \in V_t : f(x_t) = y\})$.

If $t \in A$, Predict $\phi(t)$ and update $V_{t+1} = \{f \in V_t : f(x_t) = \phi(t)\}$.

If $t \notin A$, Predict l_t and update $V_{t+1} = \{f \in V_t : f(x_t) = l_t\}$.

The number of expert we constructed is $\sum_{j=0}^{L(H)} \binom{T}{j} k^j \leq (Tk)^{L(H)}$. Denote the number of mistakes made by the expert $E_{A,\phi}$ after T rounds by $L_{A,\phi,T}$. If we apply the algorithm from [Auer et al. \(2003\)](#) with the set of experts we've constructed, we obtain an algorithm that makes at most

$$\min_{A,\phi} L_{A,\phi,T} + e \sqrt{kT L(H) \log(Tk)}$$

mistakes in expectation, whenever it runs on a sequence $z \in \mathcal{Z}^T$. To finish, we show that $\min_{A,\phi} L_{A,\phi,T} \leq \operatorname{Err}(H, z)$.

Let $f \in H$ be a function for which $\operatorname{Err}(H, z) = |\{t \in [T] : f(x_t) \notin Y_t\}|$. Denote by $A \subset [T]$ the set of rounds in which the SOA algorithm errs when running on the sequence $(x_1, f(x_1)), \dots, (x_T, f(x_T))$ and define $\phi : A \rightarrow Y$ by $\phi(t) = f(x_t)$. Since the SOA algorithm makes at most $L(H)$ mistakes, $|A| \leq L(H)$. It is not hard to see that the predictions of the expert $E_{A,\phi}$ coincide with the predictions of the expert E_f . Thus,

$$L_{A,\phi,T} = |\{t \in [T] : f(x_t) \notin Y_t\}| = \operatorname{Err}(H, z).$$

3.3 Theorem 2.4

The upper bounds follows from Equation (1), together with Theorems 2.2 and 2.3. The lower bounds follows from the corresponding bounds for the class $[k]^{[L]}$, and the fact that this class can be realized by $\mathcal{W}^{d,k}(D)$: Associate the set $[L]$ with $e_1, \dots, e_L \in \mathbb{R}^d$, the first L vectors in the standard basis of \mathbb{R}^d . A function $f : \{e_1, \dots, e_L\} \rightarrow [k]$ can be realized by the matrix $W \in \mathcal{W}^{d,k}(D)$ whose i 'th row is $\sum_{j \in [m]: f(j)=i} e_j$.

3.4 Theorem 2.5

We shall use the following Lemma. Consider the following game: A r.v., U , is sampled uniformly from Y . Then the player, that does not observe U , try to guess U . After each prediction, \hat{y}_t , he only receives an indication whether $U = \hat{y}_t$.

Claim 2 Let $R = |\{1 \leq t \leq |Y| - 1 : \hat{y}_t \neq U\}|$. Then $\mathbb{E}[R] \geq \frac{|Y|-1}{2}$.

Proof We prove the claim by induction on $|Y|$. For $|Y| = 2$ it follows from the fact that U is independent from \hat{y}_1 . For $|Y| > 2$, we note that, since U is independent from \hat{y}_1 , the probability that $\hat{y}_1 \neq U$ is at least $\frac{|Y|-1}{|Y|}$. Also, conditioned on the event that $\hat{y}_1 \neq U$, U and

$\hat{y}_2, \hat{y}_3, \dots$ satisfies the requirements of the Claim with the label set $Y \setminus \{\hat{y}_1\}$. Thus, by the induction hypothesis,

$$\mathbb{E}[R] \geq (1 + \mathbb{E}[R|\hat{y}_1 \neq U]) \cdot \Pr(\hat{y}_1 \neq U) \geq \left(1 + \frac{|Y| - 2}{2}\right) \cdot \frac{|Y| - 1}{|Y|} = \frac{|Y| - 1}{2}$$

□

Claim 3 Let H be the class of all functions $f : [\Delta] \times [k] \rightarrow [k]$ such that $f|_{\{j\} \times [k]}$ is a bijection for every $j \in [\Delta]$. Then

$$\text{B-Err}_H^r(T) \geq \Delta \cdot (k - 1) \cdot \frac{k}{4}$$

Proof Consider the following algorithm, applied by the adversary:

1. For $j = 1, 2, \dots, \Delta$
 - 1.1. For $m = 1, 2, \dots, k - 1$
 - 1.1.1. Choose $y_{j,m} \in [k] \setminus \{y_{j,1}, \dots, y_{j,m-1}\}$ uniformly at random.
 - 1.1.2. For $n = 1, \dots, k - m$
 - 1.1.2.1. Expose the learner the instance (j, m) .
 - 1.2. Let $y_{j,k}$ to be the element in the singleton $[k] \setminus \{y_{j,1}, \dots, y_{j,k-1}\}$.

By Claim 2, for every (j, m) , the adversary causes the learner to make $\geq \frac{k-m}{2}$ mistakes at the predictions for the instance (j, m) . Thus, the expected value of the total number of mistakes is $\Delta \cdot \sum_{m=1}^{k-1} \frac{(k-m)}{2} = \Delta \cdot (k-1) \cdot \frac{k}{4}$. Also, it is clear that the function $f : [\Delta] \times [k] \rightarrow [k]$ defined by $f(j, m) = y_{j,m}$ is in H . Thus, the sequence produced by the adversary is realizable by H .

□

Theorem 2.5 follows from the following claim.

Claim 4 If $D^2 \geq k^3d$ then the class H from Claim 3 with $\Delta = \lfloor \frac{d}{2} \rfloor$ is realized by $\mathcal{W}^{d,k}(D)$.

Proof Let $\bar{d} = \lfloor d/2 \rfloor$. Instead of working in \mathbb{R}^d , we work in $\mathbb{C}^{\bar{d}}$. We identify each $(j, m) \in [\bar{d}] \times [k]$ with $x_{j,m} := e^{\frac{m2\pi i}{k}} \cdot e_j$. Here, e_j is the j 'th vector in the standard basis of $\mathbb{C}^{\bar{d}}$.

Let $f \in H$. We must show that f is induced by some $W \in \mathcal{W}^{d,k}(D)$ in the sense that

$$\forall (j, m) \in [\bar{d}] \times [k], (Wx_{j,m})_{f(j,m)} \geq 1 + \max_{m' \neq m} (Wx_{j,m})_{m'} .$$

Indeed, we let $W \in M_{k,\bar{d}}(\mathbb{C})$ to be the matrix defined by

$$\forall (j, m) \in [\bar{d}] \times [k], W_{f(j,m),j} = k^2 \cdot e^{\frac{m2\pi i}{k}} .$$

We note that for every $\forall j \in [\bar{d}]$ and $m, m' \in [k]$,

$$(Wx_{j,m})_{f(j,m')} = k^2 \cdot \langle e^{\frac{m'2\pi i}{k}}, e^{\frac{m2\pi i}{k}} \rangle = k^2 \cdot \cos\left(2\pi \frac{f(j, m) - f(j, m')}{k}\right) .$$

Thus,

$$(Wx_{j,m})_{f(j,m)} \geq k^2 \cdot (1 - \cos(2\pi/k)) + \max_{m' \neq m} (Wx_{j,m})_{f(j,m')} \geq 1 + \max_{m' \neq m} (Wx_{j,m})_{m'}$$

Where the last inequality follows from the fact that by Taylor's Theorem, for $x \in [0, 2\pi/3]$,

$$1 - \cos(x) \geq \frac{x^2}{2} - \frac{x^4}{24} \geq \frac{x^2}{4}$$

□

4 Conclusion and future work

We have bounded the price of bandit information in the setting of hypothesis class based on-line learning and extended the results of Auer et al. (2003). We applied our results to estimate the bandit error rate of the class of large margin classifiers.

The focus of this paper is information theoretic. That is, we have ignored time complexity issues. It is of interest to study the *computational price of bandit information* – i.e. how the required runtime grows when moving from the full-info to the bandit scenario. It is instructive to consider the PAC setting. Given a learning algorithm, A , for a class H in the PAC full-info setting we can simply construct a bandit learning algorithm as follows – given a sample of unlabeled instances, we guess, for each instance, a label from Y , uniformly at random. Typically, we will be correct on about $\frac{1}{|Y|}$ of the examples. Thus, we can generate a *labeled* i.i.d. sample whose size is $\frac{1}{|Y|}$ -fraction of the original sample, and run the full-info algorithm A on this sample. Using this construction (see Daniely et al. (2011)), it easily follows that in the PAC setting, the price of bandit information, both information theoretic and computational, is $O(k)$. Is this true in the on-line setting as well? We note that this question is open and interesting already for the class of large-margin multiclass linear separators.

There is still some room for improvements of the bounds in Theorems 2.2 and 2.3. We conjecture that the optimal bounds are that for every class H , $B\text{-Err}_H^r(T) = O(|Y| \cdot L(H))$ and $B\text{-Err}_H^a(T) = O(\sqrt{|Y| \cdot L(H)T})$.

Theorem 2.3 together with Theorem 2.1 characterize the bandit-agnostic error rate up to a factor of $\tilde{O}(\sqrt{k})$. It is of interest to find a tighter characterization. We note that Theorem 2.1 shows that the bandit Littlestone dimension characterizes the error rate in the bandit realizable case for deterministic algorithms. It is an open question to show that this dimension quantifies the error rate also in the agnostic case and for randomized algorithms in the realizable case.

References

- P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.
- P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. The nonstochastic multiarmed bandit problem. *SICOMP: SIAM Journal on Computing*, 32(1):48–77, 2003.

- S. Ben-David, D. Pal, , and S. Shalev-Shwartz. Agnostic online learning. In *COLT*, 2009.
- Nicolò Cesa-Bianchi, Yoav Freund, David Haussler, David P. Helmbold, Robert E. Schapire, and Manfred K. Warmuth. How to use expert advice. *Journal of the Association for Computing Machinery*, 44(3):427–485, May 1997.
- K. Crammer and Y. Singer. Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research*, 3:951–991, 2003.
- V. Dani, T. Hayes, and S.M. Kakade. The price of bandit information for online optimization. *Advances in Neural Information Processing Systems*, 20:345–352, 2008.
- A. Daniely, S. Sabato, S. Ben-David, and S. Shalev-Shwartz. Multiclass learnability and the erm principle. In *COLT*, 2011.
- S.M. Kakade, S. Shalev-Shwartz, and A. Tewari. Efficient bandit algorithms for online multiclass prediction. In *International Conference on Machine Learning*, 2008.
- N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1988.
- A. Rakhlin, K. Sridharan, and A. Tewari. Online learning: Random averages, combinatorial parameters, and learnability. In *NIPS*, 2010.

The SOA algorithm

For completeness, we outline the SOA algorithm (of Ben-David et al. (2009) and Daniely et al. (2011)) for a class $H \subset Y^X$.

Algorithm 2 Standard Optimal Algorithm (SOA)

- 1: Initialize: $V_0 = H$.
 - 2: **for** $t = 1, 2, \dots$ **do**
 - 3: receive x_t .
 - 4: for $y \in Y$, let $V_t^{(y)} = \{f \in V_{t-1} : f(x_t) = y\}$.
 - 5: predict $\hat{y}_t \in \operatorname{argmax}_y L(V_t^{(y)})$.
 - 6: receive true answer y_t .
 - 7: update $V_t = V_t^{(y_t)}$.
 - 8: **end for**
-